

Содержание

1 Условие задачи.....	3
2 Математическая модель.....	4
3 Алгоритм.....	5
4 Реализация алгоритма.....	5
5 Вывод.....	8
Список литературы.....	8

1 Условие задачи

Вариант 10

Какое минимальное количество спичек необходимо для того, чтобы выложить на плоскости N квадратов со стороной в одну спичку? Спички нельзя ломать и класть друг на друга. Вершинами квадратов должны быть точки, где сходятся концы спичек, а сторонами – сами спички.

Напишите программу, которая по количеству квадратов N , которые необходимо составить, находит минимальное необходимое для этого количество спичек.

Входные данные

В единственной строке входного файла INPUT.TXT записано одно целое число N ($1 \leq N \leq 10^9$).

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести одно целое число – минимальное количество спичек, требуемых для составления заданного количества квадратов.

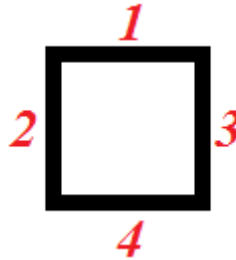
Пример

№	INPUT.TXT	OUTPUT.TXT
1	4	12

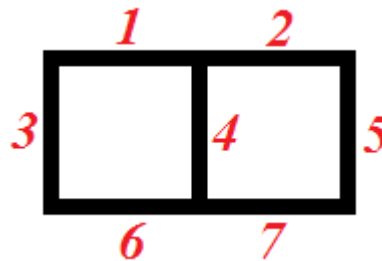
2 Математическая модель

Анализ алгоритма

Один квадрат может быть составлен из 4-х спичек



Два квадрата из 7-ми спичек



Очевидно, что квадраты следует располагать так, чтобы они образовывали прямоугольник, «близкий» к квадрату.

Пусть ширина этого прямоугольника

$$width = \lfloor \sqrt{n} \rfloor$$

тогда длина прямоугольника

$$length = \left\lfloor \frac{n}{width} \right\rfloor$$

Количество спичек для составления такого прямоугольника понадобится

$$k = width \cdot (length + 1) + length \cdot (width + 1)$$

Количество квадратов, не поместившихся в этот прямоугольник

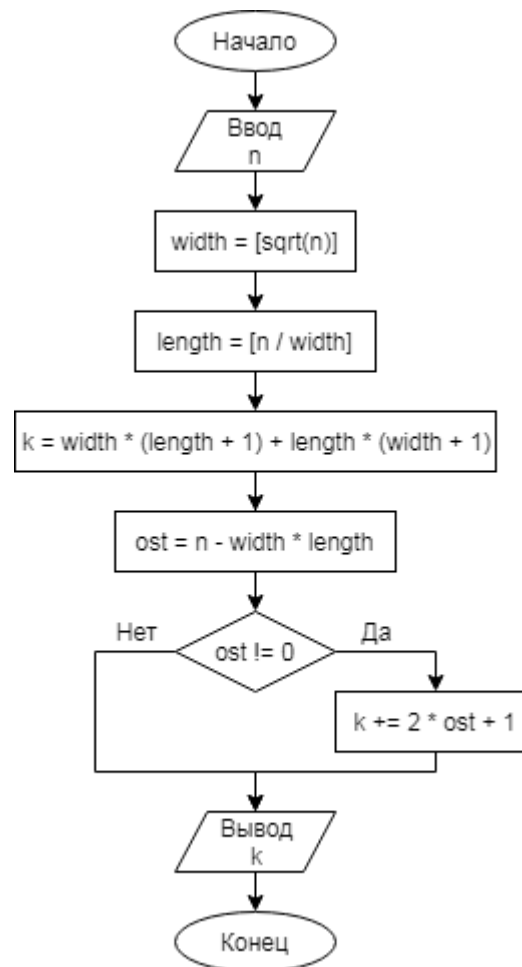
$$ost = n - width \cdot length$$

Их пристроим отдельной строкой внизу прямоугольника, на что дополнительно понадобится спичек

$$2 \cdot ost + 1$$

3 Алгоритм

Блок-схема



4 Реализация алгоритма

Program.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.IO;

namespace Project
{
    class Program
    {
        static void Main(string[] args)
        {
            Int32 n,          // количество квадратов
            k,                // количество спичек
            width,            // ширина прямоугольника
            length,           // длина прямоугольника
            ost;              // количество квадратов, не поместившихся в прямоугольник

            string line;
```

```

// считывание текстового файла input.txt
try
{
    // передаем путь к файлу и имя файла в конструктор StreamReader
    StreamReader sr = new StreamReader("input.txt");

    // читаем первую строку текста
    line = sr.ReadLine();

    // присваиваем переменной считанное значение
    n = Convert.ToInt32(line);

    // продолжаем читать, пока не дойдем до конца файла
    while( line != null )
    {
        // выводим строку в окно консоли
        Console.WriteLine("Количество квадратов = " + line + "\n");

        // читаем следующую строку
        line = sr.ReadLine();
    }

    // закрываем файл
    sr.Close();

    //-----НАЧАЛО РЕШЕНИЯ ЗАДАЧИ-----

    // ширина прямоугольника = [корень(n)]
    width = Convert.ToInt32(Math.Sqrt(n));
    Console.WriteLine("Ширина прямоугольника = " + width);

    // длина прямоугольника
    length = n / width;
    Console.WriteLine("Длина прямоугольника = " + length);

    // количество спичек для прямоугольника
    k = width * (length + 1) + length * (width + 1);
    Console.WriteLine("Количество спичек для прямоугольника = " + k);

    // количество квадратов, не поместившихся в прямоугольник
    ost = n - width * length;
    Console.WriteLine("Количество квадратов, не поместившихся в прямоугольник
= " + ost + "\n");

    // общее количество спичек
    if( ost != 0 )
        k += 2 * ost + 1;

    Console.WriteLine("Общее количество спичек = " + k);

    //-----КОНЕЦ РЕШЕНИЯ ЗАДАЧИ-----

    // запись в текстовый файл output.txt
    // передаем путь к файлу и имя файла конструктору StreamWriter
    StreamWriter sw = new StreamWriter("output.txt");

    // записываем значение k в файл
    sw.WriteLine(k);

    // закрываем файл
    sw.Close();
}
catch( Exception e )
{

```

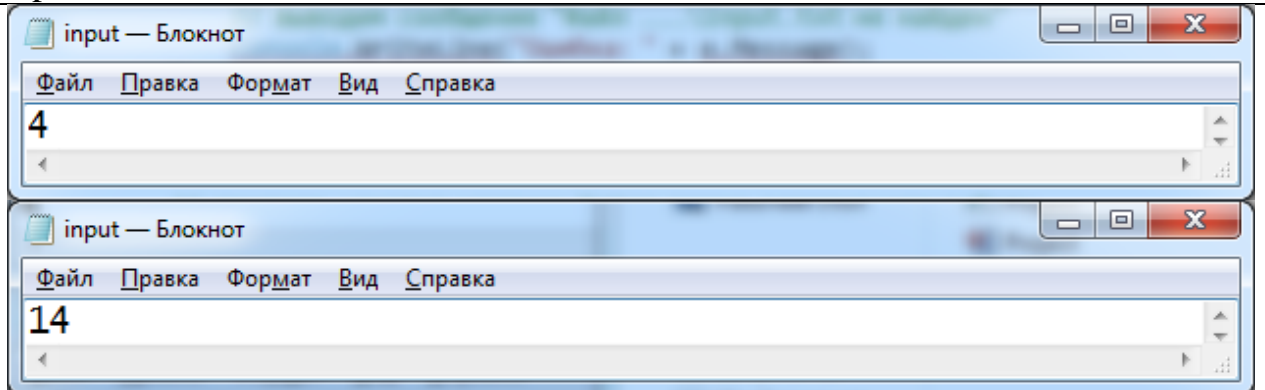
```

        // выводим сообщение "Файл ...\input.txt не найден"
        Console.WriteLine("Ошибка: " + e.Message);
    }
    finally
    {}

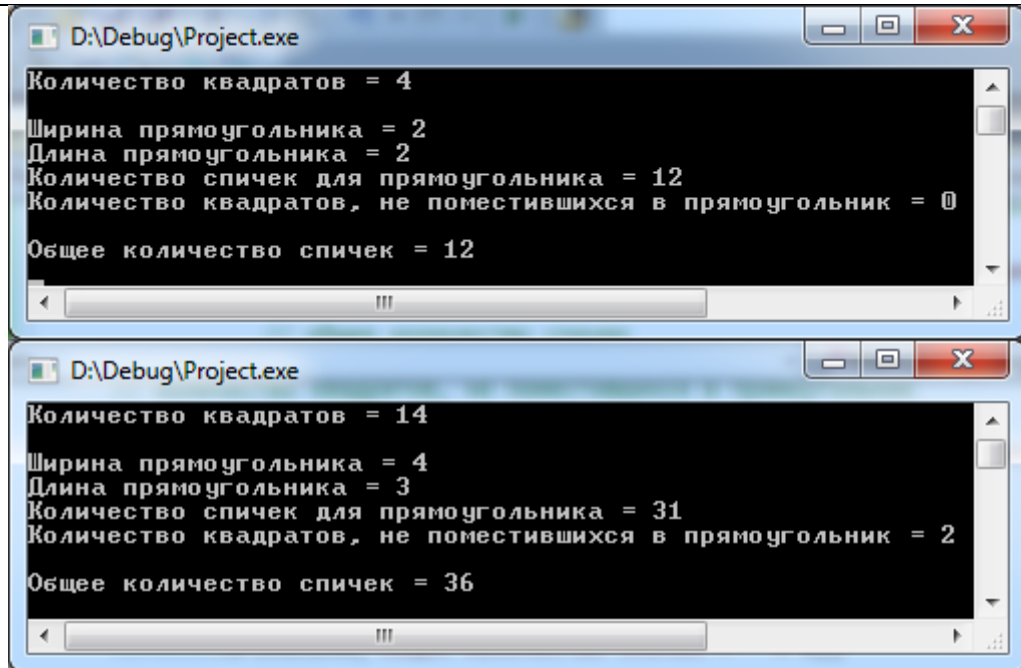
    // ожидаем нажатия клавиши Enter перед завершением работы
    Console.ReadLine();
}
}
}

```

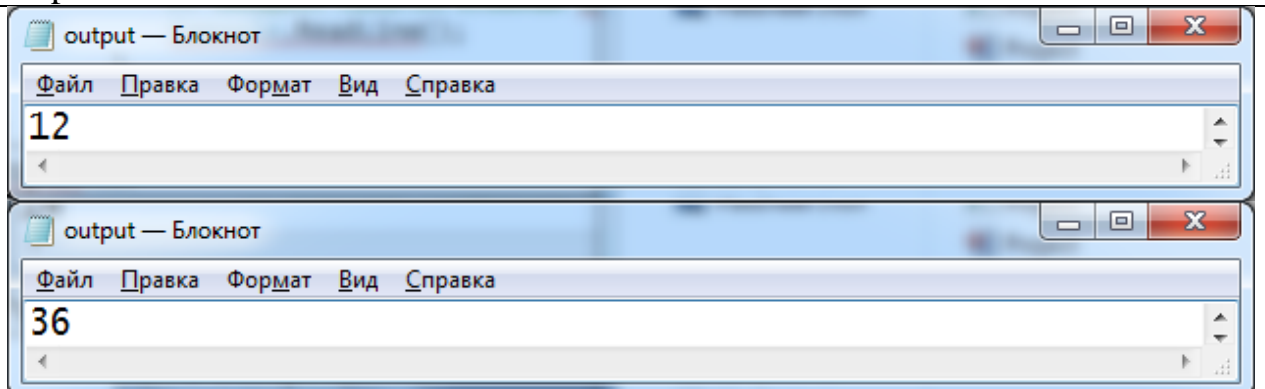
input.txt



Project.exe



output.txt



5 Вывод

В ходе решения задачи составлена математическая модель решения задачи и проведен анализ алгоритма.

Задача решена в интегрированной среде Microsoft Visual Studio C# 2010 Express.

Решение задачи проверено двумя входными данными.

Список литературы

1 Абрамян, Михаил Visual C# на примерах / Михаил Абрамян. - М.: БХВ-Петербург, 2016. - 436 с.

2 Вагнер, Билл C# Эффективное программирование / Билл Вагнер. - М.: ЛОРИ, 2017. - 320 с.

3 Культин, Никита Основы программирования в Microsoft Visual C# 2010 / Никита Культин. - М.: БХВ-Петербург, 2017. - 389 с.

4 Ник, Рендольф Visual Studio 2010 для профессионалов / Рендольф Ник. - М.: Диалектика / Вильямс, 2016. - 516 с.

5 Руководство по C# - Часть 1: https://professorweb.ru/my/csharp/charp_theory/level1/index1.php